

***Amendments to the Claims***

The listing of claims will replace all prior versions, and listings of claims in the application.

Claims 1-26. (Canceled).

Claim 27. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of the instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) a data path configured to transfer load data from the memory system to the execution unit, and

(iv) alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 28. (Previously Presented) The microprocessor according to claim 27, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 29. (Previously Presented) The microprocessor according to claim 27, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claims 30-33. (Canceled).

Claim 34. (Currently amended) A computer system, comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) a superscalar processor configured to execute the instructions, wherein the superscalar processor is configured to initiate more than one instruction in a clock cycle, the processor having,

(1) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer,

(2) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the address generation circuitry is further adapted to generate addresses for the load and store requests as soon as all operands are valid and the address generation circuitry is available for address generation.

Claim 35. (Currently amended) A computer system, comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) a superscalar processor configured to execute the instructions, wherein the superscalar processor is configured to initiate more than one instruction in a clock cycle, the processor having,

(1) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer,

(2) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the generated addresses include linear and physical addresses, and the address generation circuitry is further adapted to general physical addresses corresponding to linear addresses.

Claim 36. (Currently amended) A computer system, comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) a superscalar processor configured to execute the instructions, wherein the superscalar processor is configured to initiate more than one instruction in a clock cycle, the processor having,

(1) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer,

(2) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the load store unit includes alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary.

Claim 37. (Previously Presented) The system according to claim 36, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 38. (Previously Presented) The system according to claim 36, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claim 39-42. (Canceled).

Claim 43. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a

memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request,

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment, and

(v) alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 44. (Previously Presented) The microprocessor according to claim 43, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 45. (Previously Presented) The microprocessor according to claim 43, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claims 46-47. (Canceled).

Claim 48. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:



- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system,
  - (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request,
  - (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment, and
  - (iv) address generation circuitry adapted to generate addresses for the load and store requests when all operands are valid and the address generation circuitry is available for address generation,
- wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 49. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

- (a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and
- (b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load

request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system,
- (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request,
- (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment, and
- (iv) address generation circuitry adapted to generate linear addresses for the load and store requests, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 50. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

(i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system,

(ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request,

(iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment, and

(iv) address generation circuitry adapted to generate addresses for the load and store requests, including generation of linear addresses and corresponding physical addresses,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 51. (Canceled).

Claim 52. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

(i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system,

(ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request, and

(iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 53. (Canceled).

Claim 54. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of the instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction

from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system, and

(iii) a data path configured to transfer load data from the memory system to the execution unit,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the address generation unit is further configured to generate load and store addresses when all operands are valid and the address generation unit is available for address generation.

Claim 55. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of the instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store

requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system, and

(iii) a data path configured to transfer load data from the memory system to the execution unit,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 56. (Canceled).

Claim 57. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of the instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and

(iii) a data path configured to transfer load data from the memory system to the execution unit,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and



wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 58-59. (Canceled).

Claim 60. (Currently amended) A computer system, comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) a superscalar processor configured to execute the instructions, wherein the superscalar processor is configured to initiate more than one instruction in a clock cycle, the processor having,

(1) an instruction fetch unit configured to provide a plurality of instructions to an instruction buffer,

(2) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction

from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the load store unit is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 61. (Canceled).

Claim 62. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the address generation unit is further configured to generate load and store addresses as soon as all operands are valid and the address generation unit is available for address generation.

Claim 63. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 64. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a displacement.

Claim 65. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a

memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 66. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory



system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the load store unit is further adapted to make memory-mapped input/output (I/O) load requests in the program order.

Claim 67. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

(a) an instruction fetch unit configured to provide a plurality of the instructions to an instruction buffer; and

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address ~~may be~~ is generated out of the program order,

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system,

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request, and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions from the instruction buffer in a clock cycle and

wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 68-71. (Canceled).

Claim 72. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction

from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;
  - (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and
  - (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;
- wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and
- wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests when all operands are valid and the address generation circuitry is available for address generation.

Claim 73. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at

least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;
- (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and

- (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the execution unit further comprises address generation circuitry adapted to generate linear addresses for the load and store requests, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 74. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;
- (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and
- (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, including generation of linear addresses and corresponding physical addresses.

Claim 76. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;
  - (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and
  - (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;
- wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 77. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

- (i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;
- (ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and
- (iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the execution unit is further configured to merge data returning from the memory system with initial contents of a destination register.

Claims 78-79. (Canceled).

Claim 80. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;
- (iii) a data path configured to transfer load data from the memory system to the execution unit; and



(iv) alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle.

Claim 81. (Previously Presented) The microprocessor according to claim 80, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 82. (Previously Presented) The microprocessor according to claim 80, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claim 83-86. (Canceled).

Claim 87. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first

instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and

(iii) a data path configured to transfer load data from the memory system to the execution unit;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the address generation unit is further configured to generate load and store addresses when all operands are valid and the address generation unit is available for address generation.

Claim 88. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first

instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and
- (iii) a data path configured to transfer load data from the memory system to the execution unit;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 89. (Canceled).

Claim 90. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at

least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and
- (iii) a data path configured to transfer load data from the memory system to the execution unit;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the execution unit is further configured to merge data returning from the memory system with initial contents of a destination register.

Claim 91. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is

made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and
- (iii) a data path configured to transfer load data from the memory system to the execution unit;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the data path is further configured to merge data returning from the memory system with initial contents of a destination register.

Claims 92-93. (Canceled).

Claim 94. (Currently amended) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

- (a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the address generation circuitry is further adapted to generate addresses for the load and store requests when all operands are valid and the address generation circuitry is available for address generation.

Claim 95. (Currently amended) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,

- (i) a register file,
  - (ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and
  - (iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,
- wherein the generated addresses include linear and physical addresses, and the address circuitry is further adapted to generate physical addresses corresponding to linear addresses.

Claim 96. (Currently amended) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

- (a) a memory system configured to retain instructions and data, the instructions having a program order; and
- (b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,
  - (i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the load store unit includes alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary.

Claim 97. (Previously Presented) The microprocessor according to claim 96, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 98. (Previously Presented) The microprocessor according to claim 96, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claims 99-103. (Canceled).



Claim 104. (Currently amended) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order; and

(b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,

(i) a register file,

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,

wherein the load store unit is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 105. (Currently amended) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

- (a) a memory system configured to retain instructions and data, the instructions having a program order; and
  - (b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,
    - (i) a register file,
    - (ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order, and
    - (iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file,
- wherein the execution unit further includes merge data circuitry configured to merge data returning from the memory system with initial contents of a destination register.

Claim 106-107. (Canceled).

Claim 108. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;
- (iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request;
- (iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment; and
- (v) alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle.

Claim 109. (Previously Presented) The microprocessor according to claim 108, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 110. (Previously Presented) The microprocessor according to claim 108, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claims 111-112. (Canceled).

Claim 113. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

(i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the address generation unit is further configured to generate load and store addresses as soon as all operands are valid and the address generation unit is available for address generation.

Claim 114. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a

second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;
- (iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and
- (iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 115. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
  - (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;
  - (iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and
  - (iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;
- wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and
- wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 116. (Canceled).

Claim 117. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

- (i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;
- (ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;
- (iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and
- (iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory



system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the data path is further configured to merge data returning from the memory system with initial contents of a destination register.

Claim 118. (Currently amended) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having

(i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment;

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle and

wherein the load store unit includes merge data circuitry configured to merge data returning from the memory system with initial contents of a destination register.

Claims 119-128. (Canceled).

Claim 129. (Currently amended) In a superscalar microprocessor having an execution unit adapted to execute a plurality of instructions and to issue load instructions out-of-order, a method for managing requests for loads and stores to and from a memory device, the method comprising:

calculating an address for an instruction and transferring said address to a load store unit;

determining whether said instruction involves at least one of a load operation and a store operation;

checking, if said instruction has a load operation, for an address collision and for any write pendings, and signaling the outcome of said check;

making a request to said memory device based on a priority scheme and the results of said checking step, wherein said priority scheme includes making at least one load request out of an ordering so the one load request ~~can be~~ is made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the ordering;

receiving requested data from said load operation and/or said store operation in a data path portion of said load store unit; and

aligning said requested data if said requested data is unaligned,

wherein said step of checking includes comparing the first address of said load operation against the first and last address for an older unretired store operation.

Claims 130-135. (Canceled).

Claim 136. (Previously Presented) A method for executing one or more instructions out of order using a superscalar microprocessor, the method comprising:

receiving a plurality of instructions having an ordering, the plurality of instructions including a store instruction and a load instruction, the store instruction being before the load instruction in the ordering;

generating a load address for the load instruction and a store address for the store instruction, wherein at least one of the load address and the store address is generated out of order with respect to the ordering;

comparing the load address to the store address;

determining, in part from the comparison, if the load instruction depends on the store instruction;

if the load instruction does not depend on the store instruction, then retiring at least a portion of data provided from a data cache according to the load address, the provided data having been aligned if the load address is unaligned; and

if the load instruction does depend on the store instruction, then retiring at least a portion of load data according to store data received for the store instruction.

Claim 137. (Previously Presented) The method of claim 136, further comprising:

merging the at least a portion of data provided from the data cache with initial data from a load destination register; and

merging the at least a portion of load data according to store data with initial data from a load destination register.

Claim 138. (Previously Presented) The method of claim 136, further comprising:

writing results of the plurality of instructions into preassigned locations in a register file;

storing at least one of the load address and the store address into a first one of a plurality of address buffers; and

wherein the comparing the load address to the store address comprises receiving contents of the first address buffer.

Claim 139. (Previously Presented) The method of claim 136, further comprising preventing load bypassing of load operations that would otherwise incorrectly modify state of a system coupled to the microprocessor.

Claim 140. (Previously Presented) The method of claim 136, wherein the comparing the load address to the store address includes determining if any byte referenced by the load instruction overlaps with any byte referenced by the store instruction.